UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/802,191 | 03/17/2004 | Xuguang Yang | 14908US02 | 3425 |

23446 7590 01/28/2008
MCANDREWS HELD & MALLOY, LTD
500 WEST MADISON STREET
SUITE 3400
CHICAGO, IL 60661

| EXAMINER | |
|---|---|
| RICHARDSON, THOMAS W | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 4121 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 01/28/2008 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

| *Office Action Summary* | **Application No.** | **Applicant(s)** | |
|---|---|---|---|
| | 10/802,191 | YANG, XUGUANG | |
| | **Examiner** | **Art Unit** | |
| | THOMAS RICHARDSON | 4121 | |

***-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --***

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *17 March 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-28* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-28* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *17 March 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *24 November 2004*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

## DETAILED ACTION

Claims 1-28 are pending for examination.

Claims 1-28 are rejected.

### *Claim Objections*

1. Claims 17 and 26 objected to because of the following informalities: Minor grammatical errors.

The preamble of claim 17 reads "The wireless electronic device network of claim 6…" Examiner suggests that "wireless" be removed, as claim 6 describes an electronic device network, not a wireless network.

Claim 26 contains an apparent misspelling of the word "and." Examiner assumes the claim should read "The method according to claim 18, wherein illegal ***and*** unexpected operation codes are employed as an escape sequence during encoding."

Appropriate correction is required.

### *Claim Rejections - 35 USC § 112*

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claim 17 recites the limitation "The wireless electronic device network" in claim 6. There is insufficient antecedent basis for this limitation in the claim.

### *Claim Rejections - 35 USC § 102*

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –

> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

3. Claims 1-4, 6-9, 13-15, 17-18, and 23-28 are rejected under 35 U.S.C. 102(e) as being anticipated by US 6 671 703, Thompson et al.

4. As per claim 1, Thompson teaches a generator of difference information, the generator comprising:

a first stream of information, the first stream comprising a plurality of first bytes of data (column 3, lines 34-37, where the server compares an original file to a revision of the file);

a second stream of information, the second stream comprising a plurality of second bytes of data (column 3, lines 34-37, where the server compares an original file to a revision of the file); and

an array storing operations for tree-based encoding of the first and second streams of information, wherein the generator simultaneously traverses the first and second streams of information, analyzes the plurality of first and second bytes of data encountered in the first and second streams of information, determines difference information between the first and second streams of information, and outputs the difference information between the first and second streams of information (column 3, lines 42-57, where the method compares bytes in both files, determining if there is a mismatch, and in that case using a token to sync the files, also column 3, lines 35-43, where the server generates a delta file that shows the differences between the two files).

5. As per claim 2, Thompson teaches the generator according to claim 1, further comprising a differencing component for computing and expressing the difference information employing a differencing instruction set (column 3, lines 35-43, where the server generates a delta file that shows the differences between the two files).

6. As per claim 3, Thompson teaches the generator according to claim 2, wherein the differencing instruction set comprises at least one operation selected from a match operation, an insert operation, a delete operation, and a replace operation (column 3, lines 48-57, where the sync determines if there was an insert, delete, or replace action taken on the file).

7. As per claim 4, Thompson teaches the generator according to claim 1, further comprising an encoder providing tree-based encoding, the encoder employing a block-based hierarchical representation, and the encoder segmenting blocks during encoding (column 3, lines 45-57, where the token comprises a number of bytes, therefore represents a segmented block of the larger data program block).

8. As per claim 6, Thompson teaches an electronic device network adapted to dispense streaming updates to at least one of a plurality of electronic devices, the updates for updating one of firmware and software (column 3, lines 17-32, where the server updates a file at a remote client), the electronic device network comprising:

a generator generating streaming updates, the generator processing at least one of a plurality of blocks of content, the at least one of a plurality of blocks of content comprising a stream of bytes, the generator processing the at least one of a plurality of blocks of content until reaching an end of the stream of bytes (column 3, lines 32-38, where the server compares an original file to a revised file);

a server communicatively coupled to the at least one of a plurality of electronic devices, the server disseminating the streaming updates to the at least one of a plurality of electronic devices (Figure 1, where the server 14 distributes the updated information); and

a processor in the at least one of a plurality of electronic devices for processing the streaming updates received from the server (Figure 1, where the clients are computers, therefore have processors).

9. As per claim 7, Thompson teaches the electronic device network according to claim 6, wherein the generator employs an array to store operations used to transform a first stream of information into a second stream of information (Figure 2, where the server contains a memory with the file difference synchronization system), the generator processing the first stream and the second stream in a byte-by-byte fashion to generate streaming updates (column 3, lines 42-57, where the method compares bytes in both files, determining if there is a mismatch, and in that case using a token to sync the files), and each byte is one of a text character and a binary value of at least one of the first and second streams (column 3, lines 42-57, where the method compares bytes in both files, determining if there is a mismatch, and in that case using a token to sync the files. It is well known in the art that bytes in a computer system are made up of binary bits).

10. As per claim 8, Thompson teaches the electronic device network according to claim 6, wherein the generator maintains a transform array wherein a minimum weight is assigned to a set of operations, the minimum weight being computed by employing an edit distance computation in management of an operational array (column 8, lines 42-47, where a traversal routine is run computing the least cost path to a node).

11. As per claim 9, Thompson teaches the electronic device network according to claim 8, wherein the set of operations comprises at least one of a replace operation, a match operation, an insert operation, and a delete operation (column 8, lines 48-60, where the traversal routine is for a delete action).

12. As per claim 13, Thompson teaches the electronic device network according to claim 6, wherein the electronic device network is one of a wired and a wireless network (Figure 1, where the server communicated with remote devices on a wired or wireless network, those devices being a desktop, laptop, and PDA).

13. As per claim 14, Thompson teaches the electronic device network according to claim 6, wherein the streaming updates comprise a difference output for two streams comprising one of binary data and text data (column 3, lines 34-38, where the server sends a delta modification file to a remote device for updates), the difference output comprising at least one of a tree map, operational codes for operations comprising at least one of a replace operation, a match operation, a delete operation, and an insert operation (column 4, lines 1-5, where the operations are skip, delete, insert, and replace), and data characters associated with at least one of the insert operation and the replace operation (column 3, lines 45-57, where a token associated with the operations is grabbed).

14. As per claim 15, Thompson teaches the electronic device network according to claim 6, wherein a first stream of information and a second stream of information are processed by the generator, and wherein an operational array is computed in the generator by consuming each of the streams in small chunks, wherein a small chunk comprises one of a 64-byte block of information, a 16-byte block of information, and a 4-byte block of information (column 4, lines 37-43, where the use of Intel 80x86 or Pentium chips (well

known to have 32-bit processors) is described. Therefore, it is inherent that 4 byte (32 bit) blocks could be used in the processor of the system).

15. As per claim 17, Thompson teaches the ***wireless*** electronic device network according to claim 6, wherein the electronic device comprises at least one of a plurality of mobile electronic devices, and wherein the plurality of mobile electronic devices comprise at least one of a mobile cellular phone handset, a personal digital assistant, a pager, a multimedia device, and a camera (Figure 1, where a remote client is a PDA).

16. As per claim 18, Thompson teaches a method of generating streaming updates by converting a first stream of information into a second stream of information for updating an electronic device, the method comprising:

identifying the first and second streams of information (Figure 2, where the original file and revision file represent the two streams of information);

accessing the first and second streams of information (column 3, lines 32-42, where the first and second streams are compared to create a delta modification file);

retrieving one block of content at a time from each of the first and the second streams of information (column 3, lines 42-46, where the server goes through the files byte by byte);

determining a transform operation (column 3, lines 32-42, where the first and second streams are compared to create a delta modification file); and

executing the transform operation (column 3, lines 32-42, where the first and second streams are compared to create a delta modification file).

17. As per claim 23, Thompson teaches the method according to claim 18, wherein the update facilitates conversion of the first stream of information into the second stream

of information, wherein retrieving blocks of content from the second stream of information is performed at a fixed pace using a fixed block size, and wherein retrieving blocks of content from the first stream of information is performed at a variable pace using a variable block size, wherein a reference to the second stream of information is maintained and a cumulative offset is computed (column 3, lines 42-57, where the system compares bytes from each data string. The size and pace of processing does not matter, as the bytes are compared to each other individually and not as a stream of information).

18. As per claim 24, Thompson teaches the method according to claim 18, wherein a look-ahead operation is executed as part of retrieving blocks of content, the look-ahead operation employing data from the first and second streams of information to compute an operational array (column 3, lines 43-57, where the method will continue counting matching bytes until there is a mismatch, and does not perform an action until matching information for the next byte is determined).

19. As per claim 25, Thompson teaches the method according to claim 18, wherein a longest common sub-string technique is employed prior to determining a transform operation (column 3, lines 43-57, where the method will continue counting matching bytes until there is a mismatch, then begin taking tokens and employing differentiation methods).

20. As per claim 26, Thompson teaches the method according to claim 18, wherein illegal ***an*** unexpected operation codes are employed as an escape sequence during encoding (column 4, lines 1-5 state that there are 4 types of records. It is inherent that if an unrecognized code is presented, an error would be given and the data would not be processed).

21. As per claim 27, Thompson teaches the method according to claim 26, wherein a combination of the escape sequence, a type field of two bits, and a length representing a repetition of data associated with the type field is employed by to encode long strings of complete matches between the first and second streams of information (column 4, lines 1-17, where the skip record shows the number of bytes that are matches between the first and second strings).

22. As per claim 28, Thompson teaches the method according to claim 18, wherein the electronic device comprises at least one of a plurality of mobile electronic devices, and wherein the plurality of mobile electronic devices comprise at least one of a mobile cellular phone handset, a personal digital assistant, a pager, a multimedia device, and a camera (Figure 1, where the remote devices are a desktop, laptop, and PDA).

### ***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* v. *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

23. Claims 5, 10-12, 16, and 19-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 6 671 703, Thompson et al as applied to claims 4, 6, 15, and 18 above, and further in view of US 5 973 626, Berger et al.

24. As per claim 5, Thompson teaches the generator according to claim 4. Thompson does not teach a specific type of encoding to use when compressing difference information. Berger teaches a method and system for byte-based prefix encoding wherein:

the encoder employs variable length encoding techniques for operators in a set of operations (abstract, where the Huffman code uses variable length encoding), the encoder employing tree-based variable sized blocks (Figure 2, where the tree is used to determine variable sizes of the prefix), and wherein the generator computes a cumulative address offset (column 3, lines 11-24, where the code in the table represents a number of offset bits for each symbol).

It would have been obvious to one of ordinary skill in the art at the time of invention to use a byte-based encoding technique such as that taught by Berger in a compression and update system such as that taught by Thompson. Berger uses the Huffman code for encoding data to accelerate processing associated with encoding (abstract), which would benefit any difference generation device, as it would allow the process to be completed more efficiently. This would benefit Thompson's system especially, as the difference information between the two data streams could be encoded for faster and more efficient updating of the software elements.

25. As per claim 10, Thompson teaches the electronic device network according to claim 6, further assigning values to each operation in a set of operations (column 4, lines 1-16, where there are 4 different operations that define the delta modifications file). Thompson does not teach a specific type of encoding to use when compressing difference information. Berger teaches a method and system for byte-based prefix encoding wherein:

the generator comprises an encoder for encoding operations, the encoder assigning values to each operation in a set of operations (abstract, where the Huffman code uses variable length encoding).

It would have been obvious to one of ordinary skill in the art at the time of invention to use a byte-based encoding technique such as that taught by Berger in a compression and update system such as that taught by Thompson. Berger uses the Huffman code for encoding data to accelerate processing associated with encoding (abstract), which would benefit any difference generation device, as it would allow the process to be completed more efficiently. This would benefit Thompson's system especially, as the difference information between the two data streams could be encoded for faster and more efficient updating of the software elements.

26. As per claim 11, the combination of Thompson and Berger teaches the electronic device network according to claim 10, wherein the encoder assigns a minimum weight, wherein the minimum weight is computed by employing appropriate weights in management of a transform array (Thompson teaches this limitation. Column 8, lines 42-47, where a traversal routine is run computing the least cost path to a node).

27. As per claim 12, the combination of Thompson and Berger teaches the electronic device network according to claim 11, wherein the encoder assigns numeric values to each operation in the set of operations, wherein non-zero values are assigned to replace operators and insert operators, and zero is assigned to match operators (Thompson teaches this limitation. Column 3, lines 42-57, where a token's worth of bytes are grabbed from the file. Thus, if an insert, delete or replace function has taken place, the token will have a value. If the bytes match and the skip function is utilized, no token is present, and the value of the nonexistent token is therefore zero).

28. As per claim 16, Thompson teaches the electronic device network according to claim 15, wherein the after the server defines an operation, it reorients to a corresponding point in each of the streams to start additional encoding of a next small chunk (column 4, lines 6-17, where the server reorients according to the byte offset presented with each action)

Thompson does not teach a specific type of encoding to use when compressing difference information. Berger teaches a method and system for byte-based prefix encoding wherein:

each small chunk of each of the streams is used to define a portion of the operational array is encoded (abstract, where the Huffman code uses variable length encoding).

It would have been obvious to one of ordinary skill in the art at the time of invention to use a byte-based encoding technique such as that taught by Berger in a compression and update system such as that taught by Thompson. Berger uses the Huffman code for encoding data to accelerate processing associated with encoding (abstract), which would

benefit any difference generation device, as it would allow the process to be completed more efficiently. This would benefit Thompson's system especially, as the difference information between the two data streams could be encoded for faster and more efficient updating of the software elements.

29. As per claim 19, Thompson teaches the method according to claim 18, further comprising:

computing an output from the transform operation (column 3, lines 32-42, where the first and second streams are compared to create a delta modification file);

creating a tree-based transform output from operators determined in the transform (column 6, lines 50-55, where there is a difference tree);

outputting difference information into at least one memory structure (Figure 2, where the difference file maintains the difference information).

Thompson does not teach a specific type of encoding to use when compressing difference information. Berger teaches a method and system for byte-based prefix encoding wherein:

encoding the tree-based transform output employing at least one of variable length encoding and fixed length encoding (abstract, where the Huffman code uses variable length encoding).

It would have been obvious to one of ordinary skill in the art at the time of invention to use a byte-based encoding technique such as that taught by Berger in a compression and update system such as that taught by Thompson. Berger uses the Huffman code for encoding data to accelerate processing associated with encoding (abstract), which would benefit any difference generation device, as it would allow the process to be completed

more efficiently. This would benefit Thompson's system especially, as the difference information between the two data streams could be encoded for faster and more efficient updating of the software elements.

30. As per claim 20, the combination of Thompson and Berger teaches the method according to claim 19, further comprising:

determining whether additional blocks of content are to be processed by evaluating the first and second streams of information (Thompson teaches this limitation. Column 6, lines 46-54, where the file difference synchronization continues until and end of file (EOF) is reached);

retrieving an additional block of content from each of the first and the second streams of information upon determining that additional blocks of content are to be encoded (Thompson teaches this limitation. Column 6, lines 46-54, where the file difference synchronization continues if no end of file (EOF) is reached); and

continuing encoding until reaching an end of a stream of blocks of content to be encoded (Thompson teaches this limitation. Column 6, lines 46-54, where the file difference synchronization continues until and end of file (EOF) is reached).

31. As per claim 21, the combination of Thompson and Berger teaches the method according to claim 20, further comprising:

compressing difference information output (Thompson teaches this limitation. Column 3, lines 36-42, where the server generates a delta modification file, which gives the only the difference information, not the entire modified file); and

packaging the difference information output into an update (Thompson teaches this limitation. Column 3, lines 36-42, where the delta modification file is transmitted to the remote user).

32. As per claim 22, Thompson teaches the method according to claim 18, further comprising:

buffering content from the first stream of information and the second stream information to determine the difference information (Figure 2, where both the original and revision files are stored in the server memory); and

outputting the difference information (column 3, lines 35-42, where the delta modification file is generated by the server and sent to the remote user)

Thompson does not teach a specific type of encoding to use when compressing difference information. Berger teaches a method and system for byte-based prefix encoding wherein:

encoding the information before outputting the information (abstract, where the Huffman code uses variable length encoding).

It would have been obvious to one of ordinary skill in the art at the time of invention to use a byte-based encoding technique such as that taught by Berger in a compression and update system such as that taught by Thompson. Berger uses the Huffman code for encoding data to accelerate processing associated with encoding (abstract), which would benefit any difference generation device, as it would allow the process to be completed more efficiently. This would benefit Thompson's system especially, as the difference information between the two data streams could be encoded for faster and more efficient updating of the software elements.

### *Conclusion*

33. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

US 7 274 671, Hu teaches bitwise encoding using prefix predication

US 7 188 214, Kasriel et al teaches efficient compression using differential caching.

US 6 999 976, Abdallah et al teaches a method, apparatus, and program for using Java archive to encode file system deltas.

US 5 864 687, Proctor et al teaches a video encoder/decoder.

US 2003/0177255, Yun teaches an encoding and decoding system transmitting streaming video to wireless devices.

US 2003/0212742, Hochmuth et al teaches a method, node, and network for compressing and sending composite images.

US 2002/0091807, Goodman teaches an automatic update of firmware in processor nodes.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to THOMAS RICHARDSON whose telephone number is (571)270-1191. The examiner can normally be reached on Monday through Thursday, 8am-5pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Taghi Arani can be reached on (571) 272-3787. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

TR

/Taghi T. Arani/
Supervisory Patent Examiner, Art Unit 4121
1/23/2008